XMSF Overlay Multicast
Status Report

Mark Pullen and Dennis Moen
George Mason University
{dmoen,mpullen}@gmu.edu

Network Service Requirements for
Real Time Distributed Virtual Simulation
-XMSF Report, 2002

- Network Quality of Service (QoS)
  - end-to-end capacity, latency, jitter, and packet loss in a statistical sense
- Multicast
  - many-to-many group communication
- Reliable Multicast Transport
  - high confidence of delivery
- End-to-end network status and performance monitoring
  - need to know what the network is doing for you
- Multi-sensor systems
  - must manage streaming data with low latency
What is multicasting?

- Multicasting: sending a packet to a group of addresses in a network
  - broadcast: sending to all addresses - impractical in WAN
- IPmc allows packets to be sent to “group” addresses
  - delivered in parallel
  - “class D” address 224.0.0.0 and greater
  - LANs have inherent broadcast/multicast capability
- In WAN, set of all nodes in a multicast group forms a tree
  - grows from the node that starts the group
- Routers at forks in tree must replicate packets
  - commercial routers available that perform this function
- Multicast requires a different routing protocol from unicast
  - routing creates the multicast tree

Network with Multicast Tree

*the network duplicates the packets*
IP Multicast Directionality

• IPmc uses a many-to-many model of packet delivery
  – equivalent to full duplex for multicast
  – any participating host can send to the group
  – useful for collaborative style applications
  – a very appealing model but increases complexity of the network layer considerably

• IETF recently developed an approach called Source Specific Multicast (SSM) that uses a one-to-many model
  – equivalent to half duplex for multicast
  – only one host can send to the group
  – useful for presentation-style applications

One-to-Many Multicast
Internet Multicast Services Today

- IP multicast over the Internet not widely deployed
- IETF focus is on one-to-many multicast
- Commercial viability lacking for IP multicast across the Internet
- Result: interest in multicast based in end systems not network
  - End-to-end argument: push complexity up the stack
  - Example: TCP is complex, IP is simple

Application Layer Multicasting

Network EducationWare System
Overlay Multicast
provided by a host-based middle layer

IP Multicast tree:

XOM Overlay
### XOM Architectural Layers

<table>
<thead>
<tr>
<th>Generic Class Definition Interface (SRMP Example)</th>
<th>Routing Table</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Routing</strong></td>
<td></td>
</tr>
<tr>
<td>Group Management</td>
<td></td>
</tr>
<tr>
<td>Register</td>
<td></td>
</tr>
<tr>
<td>Join/leave</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>Path Management</td>
<td></td>
</tr>
<tr>
<td>Capacity/latency</td>
<td></td>
</tr>
<tr>
<td>Node Demand</td>
<td></td>
</tr>
<tr>
<td>Path Optimization</td>
<td></td>
</tr>
<tr>
<td>Packet Send/Receive Distribute Messages</td>
<td>Listen to Ports</td>
</tr>
<tr>
<td>Class QoS/Queueing</td>
<td></td>
</tr>
</tbody>
</table>

| UDP                                              |
| IP                                               |

© 2004 J. Mark Pullen

### XOM Group Membership

G₂ = {B, C, D}

Application B sending implies routing to group G₁ = (G₁ ∪ G₂)

© 2004 J. Mark Pullen
Group Aggregation Overlay
(Optimum Path Overlay)

Multicast Groups

<table>
<thead>
<tr>
<th>Group</th>
<th>Members</th>
</tr>
</thead>
<tbody>
<tr>
<td>g₀</td>
<td>XOM₁,₂,₃,₄</td>
</tr>
<tr>
<td>g₁</td>
<td>XOM₁,₂,₃,₄</td>
</tr>
<tr>
<td>g₂</td>
<td>XOM₁,₂,₃</td>
</tr>
<tr>
<td>g₃</td>
<td>XOM₁,₂</td>
</tr>
</tbody>
</table>

Aggregate Trees

<table>
<thead>
<tr>
<th>Tree</th>
<th>Tree Links (arcs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T₀</td>
<td>1-4, 4-2, 4-3</td>
</tr>
</tbody>
</table>

Groups g₀, g₁, g₂, g₃ share one aggregate tree T₀. T₀ is a perfect match for g₀ and g₁, but is a leaky match for g₂ and g₃. Trades off path utilization inefficiency for lower path management overhead.

XOM Functional Model
Prototype Test Scenario
Prototype Lab Test Scenarios

- With DIS-like traffic:
  - Java version 3000 messages per second
  - C++ version 8000 messages per second
Setting Up the XOM Prototype

- One host per subnet runs XOM
  - Also can run applications
  - But for highest performance a separate host is best
- Current prototype does not support a registry
  - Each XOM must know the IP address of all others
- The XOMs exchange routing data and create a separate tree per source XOM
- Two configurations available
  1. Pure Java
  2. Router module assisted by C++ (better performance)
- Multicast applications on LAN hosts run without modification
  - XOM is transparent except in received packet “from” address

XOM Prototype Command Line Parameters

- registryAddress: 0 for now
- numberOfMulticastGroups: count of groups/ports we will support
- lowestMCAddress: first group address to multicast from the subnet, dotted decimal notation (other addresses follow in sequence)
- lowest port: first UDP port to multicast (each address will get one port, in sequence)
- routingUpdateInterval: time in ms between routing updates
- useTCP: 0 for now
- partnerHostAddresses: up to 20 XOM host IP addresses in dotted decimal notation; these partners will be used without checking registry
Conclusions and Future Work

Initial results indicate overlay networking is a promising strategy for providing many-to-many multicast in the open internetwork environment of XMSF.

We are working on an architecture specification based on the properties of distributed simulation traffic plus recent networking research.

A working experimental prototype is available on our website. We plan to enhance this with better performance display.

NPS is working on a Web-service-based registry and routing information system.

Our system will be demonstrated at I/ITSEC, supporting a multicast HLA federation and Web Service-based Interest Management (WSIM).

Publications

Available on http://netlab.gmu.edu--


